

Introduction

General Introduction to the Project

- Purpose
- Introduction

Overview of Koha (and Ubuntu)

- Overview
- Koha's Iterations

Support and Resources

Prerequisites to Installation

- Hardware & Software
- Knowledge

General to the Introduction to the Project

Purpose

To successfully install the Koha integrated library system on a Linux server and to create an installation tutorial focused on full explanations and simplicity.

Introduction

The benefits of choosing a Koha installation as an independent study has many professional ramifications and all of them were factors in the decision to actually do the project. I have also enjoyed the idea of a senior thesis. A senior thesis was required for my undergraduate degree and I had a wonderful time not only in the actual process but also the feeling of accomplishment. This project of completing a Koha installation, the open-source integrated library system (ILS), not only created a senior project opportunity but it also combined a need to create a new set of skills that I find interesting and useful for my career. These skills include a general server knowledge, familiarity with web based technologies, writing and experience using an ILS.

I believe in study for its own sake but studying servers, and Information Technology (IT) in general, is useful for many reasons. With the outsourcing of IT from the hands of some librarians, I will most likely use this server and general IT knowledge to better communicate with an IT department from a non-IT position. If technological communication is the extent of the use of this project then I consider it to still be well worth while. However, in a Public Library setting there might not be someone on hand

or money to pay someone to create a print or file server or to re-image hard drives or diagnose computer issues which this project has greatly helped me to understand. I was comfortable with computers before but exposure to Linux, and servers in general, has expanded my repertoire to not only be a better rounded Librarian but also a good base knowledge for becoming an IT professional or systems librarian. This server knowledge will also help me be a better web developer being able to create my own professional development environment with ease.

I have studied web-based technology for two years now but never anything as complicated as Koha. Granted I will not be creating new modules or large modifications for Koha, but with the knowledge that I have gained of databases, how to install a web server for development purposes and the comfort I have gained with complex web applications, I am more marketable as a librarian and better able to create and modify web-based applications.

Finally, ILS experience is required for almost every library job that I have seen in the past two years. Usually, the ideal candidate will have had experience with the particular ILS used by the particular library, but experience with any ILS is at least helpful. CW/MARS is rumored to be planning on switching the ILS they provide to their customers from Innovative Interfaces' Millennium to Koha at some point in the future. This switch to Koha puts me in an excellent position not only to help colleagues to understand the Koha system as a whole but also how it operates on a day to day basis.

Justification for this project is my least concern because it is so technologically related. Technology is only getting more and more complicated and more and more difficult to understand. Keeping up with technology takes time but those who put in the time are much more likely not only to keep up with current technologies but be prepared for the new as well. With this project, I have instilled a deeper desire to study tech topics.

Overview of Koha (and Ubuntu)

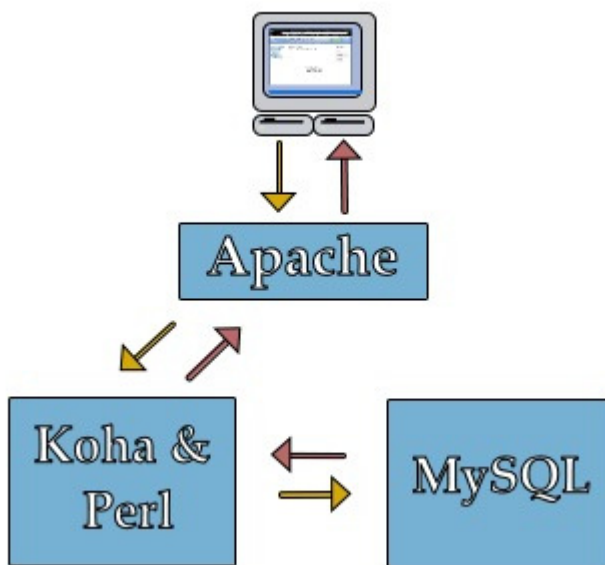
Overview

Koha is an Integrated Library System (ILS) that is freely available to the public. It is used to automate library functions from checking books in and out and creating library cards to more administrative tasks such as statistical work and setting up branch libraries. It is a powerful system which, when combined with the powerful search and indexing engine, Zebra, can quickly retrieve MARC records from a database with tens of thousands of records.¹ It is truly a free, full function, ILS and is a rising star, but currently not the most popular ILS on the market. There are more than 500 libraries including academic, public and special libraries that use Koha as their ILS including such large libraries as Delhi Public Libraries, Antioch University as well as a major Koha consortium in Vermont called VOKAL (Vermont Organization of Koha Automated Libraries).²

¹ Sirohi & Gupta, 2010

² Helling, 2010, p. 1; Delhi Public Library from Sirohi & Gupta, 2010 About the Authors; other libraries from Sirohi & Gupta, 2010, About the Authors.

This tutorial will take you through the process of installing Koha on a Linux-based system but an overview of what you will actually be doing will be helpful as a written roadmap perhaps as a companion to the table of contents. The beginning of the tutorial explains how you can install and test a web-server (Apache2 and MySQL) on your Ubuntu-Linux (Ubuntu-desktop version) system. It will then go over how to install all of the necessary Perl modules that Koha needs to run and then will help you actually install Koha itself. After the actual installation of Koha onto the system, it will then walk you through post-installation topics such as configuring Apache2 to work with Koha and setting up Koha's database. If you look at the graphic below, it tries to simply explain how Linux, the web-server, the Perl modules and Koha all interact with each other.



The user is at the top and imagine it is a catalog search for 'Herodotus'. The web browser sends the request (orange) to the server to the server where Apache picks up the request and sends it to the Koha/Perl software which. Koha sees that it is a catalog search and sends it to the catalog that resides in MySQL, Koha's database management tool of choice. MySQL performs the search for 'Herodotus' and returns the any results (red) to Koha which packages them in a nice web format and sends them through Apache to the user. It is important to note that MySQL houses all of the data for Koha including all of the book records, patron data, use statistics, etc.

You can complete an installation of the above system as you read through this tutorial. There may be many pages to go but the experience that can be derived is valuable not only because of the knowledge acquired but the familiarity and confidence to work with computers.

Warning: this tutorial is not designed for anyone to create a production system of the Koha ILS. There are many missing safety measures necessary in a production system that are not explained the most important of which is data security. In order to create a production system that is secure enough to keep safe user data such as names, addresses and phone numbers, you should consult professional IT staff or firms. Other needs for a production system are the Zebra search engine, a very specific hard drive configuration to name a few all of which are not included in this tutorial.

Koha's Iterations

Like Linux itself, there is more than one iteration of Koha. Linux has distributions like Ubuntu, Debian and OpenSUS and there are two Koha iterations are the Koha Community version (the original) and Liblime Koha. Both are free and open-source and can be customized to the library in terms of creating new features or modules. The major difference between the two is that the Community version is controlled by a board of trustees with a charter to guide development.³ Liblime Koha is controlled by a private company.⁴ The fact that one is run by a private company and the other a board of trustees does not mean that either is better in terms of functionality, usability or good for the community. Liblime only wants to sell support and hosting but the code for Liblime Koha is free.⁵ Both are safe and good to use.

Technically, there is a third iteration of Koha called Liblime Academic Koha. It is a production version meaning that there are libraries that use it as their ILS but it is currently not available to the public. Liblime and a consortium of Libraries have created this iteration and it is only available to those libraries that join the consortium or pay a percentage of the cost of its creation.⁶

Support and Resources

If you do not want to purchase books written by experts, then the best resources for Ubuntu and Koha are the community websites found at:

koha-community.org
ubuntu.com/community

These websites contain what is called the 'software documentation' which essentially amounts to manuals on how to use the software. You can search, browse or even post questions to forums about anything and everything. Both websites contain **all** of the information that you could need about the functionalities of the software. The Ubuntu Community website in particular is truly amazing in that it is moderated by real users and not wholly by the company that controls Ubuntu and its iterations. There are also thousands of websites about Linux, Ubuntu and to a certain extent Koha as well.

If you prefer, there are also many books available on Ubuntu-desktop and the server version though, again, this tutorial is designed for the desktop version of Ubuntu. The one purchased for this project was *Ubuntu Unleashed* by Helmke, Hudson and Hudson (new 2012 edition covering Ubuntu 12.04). It is a good overview text but lacked depth and examples needed to convey much of what you

³ Koha Library Software Community, 2011

⁴ Liblime, 2011

⁵ Liblime, 2011

⁶ Jones, 2011

need to know to be a Linux administrator. There is also the *Official Ubuntu Book* by Hill, Helmke, Graner and Burger with good reviews based on Amazon.com's users.

For Koha, the book purchased for this tutorial was *Koha3 Library Management System* by Sirohi and Gupta. It is a good text for those who are very comfortable with Linux but was designed for Linux administrators as the introduction will tell you up front.⁷ I found it very useful in that I had tried installing the system many times before the book was purchased and it tied together many of the information threads that I had gathered from the many online tutorials. These many online tutorials can be found by doing a Google search but a few include:

Blazing Moon: <http://www.blazingmoon.org/news/2008/12/installing-koha-3-on-ubuntu-810-a-tutorial/>

Bot Skool: <http://www.botskool.com/programming-tutorials/linux/installing-koha-ubuntu-904-910-1004>

The most important thing to remember when troubleshooting a problem is Google. You may find the solution to an issue in a book or on the documentation websites but it is far easier to plug a few search terms into the Google search box and sort through for the information needed. Using Google will not only allow you to find information faster but it also has a much broader base of information than any book on technology. With each issue that you come across you will become more and more apt not only to search correctly the first time for resolutions to the issue but, when you install the system over again (for fun or out of need), you will be likely to remember the pitfall and carefully cross the precipice the next time around.

For scholarly resources and recent publications on Koha, see Appendix C for a list of resources.

Prerequisites to Installation

Hardware & Software

Before starting the Koha installation, be sure that your computer meets the system requirements for both Ubuntu Desktop as well as Koha. However, the requirements for Ubuntu are more intense than Koha so the only requirements with which we should be concerned are those for Ubuntu Desktop. Even these requirements can be met by a good computer from about 2000:

- 1 GHz processor
- 1 Gigabyte of RAM
- 15 GB of hard drive space
- Monitor with at least 1024 by 768 screen resolution
- A USB port or a CD/DVD ROM

⁷ Sirohi & Gupta, 2010, p. 1 and back cover

Internet Access⁸

Just to give you an idea of how simple these hardware requirements are, it is impossible to find a CPU that is less than 1.6 GHz in the retail market today, Windows 7 compatible computers generally have 4-8 gigabytes of RAM and the smallest full-sized hard drive in the retail market available is around 40 GB. However, it is a good idea to have a hard drive on the larger size for a production system because the database of MARC records will be contained on that one hard drive. We are not only concerned about MARC records but also patron data, checkout history, use statistics and most importantly database indexes.⁹

For software, the only requirement to begin is a copy of Ubuntu-desktop. It is freely available online in many locations including the Ubuntu website. A good way of getting Ubuntu is to download a small application that helps you to load the operating system onto a small USB drive available at www.pendrivelinux.com. This website, and its software, allow you to put Ubuntu-desktop on a USB drive which allows a portable and easy way to install operating systems.

Knowledge

In terms of knowledge requirements for installing Koha, you should have at least a working knowledge of Linux and preferable with Ubuntu. All of the prerequisite knowledge can easily be gained with casual use of Ubuntu beforehand. It would be a good idea to learn the equivalents of Ubuntu to your own native operating system and develop a general comfort with the following tasks and concepts:

1. Command-line
2. Installation of Linux packages
3. Installation of Ubuntu-desktop from scratch

One and three are more important especially considering there is a small walk-through on how to install Linux-packages. The above concepts can be difficult to master but there are many tutorials that can be studied. For example, a simple search on YouTube brings up several videos of varying length on how to install Ubuntu on a computer including ways to have both Windows and Ubuntu on one computer at the same time.

⁸ Canonical Ltd., 2011, Installation/SystemRequirements

⁹ Database indexes are snapshots of the most used columns in a database that help the database return queries faster.

Preparing for Installation: Prerequisites

Tips and Root Access

- Tips
- Root Access
- Packages

LAMP (Linux, Apache, MySQL and PHP)

- Overview
- LAMP Installation (manual and automatic)
 - Manual Instructions
 - Automatic Instructions Using Taskset
- Extra Packages for MySQL
- Testing the -AMP Installation
- MySQL Setup

Where and How to Download Koha

Perl Modules

- Perl Module Overview
- Module Installation
 - Automatic Perl Modules Installation
 - Manual Perl Modules Installation

Tips and Root Access

Tips

Starting with a fresh installation of Ubuntu is a great way to start but not required. However, it is important to know that when you install Ubuntu it will ask you to create a first or administrative user. This tutorial assumes that you have named that first administrative user 'koha.'¹⁰ Be sure to write down

¹⁰ A fresh installation is not required as long as you can create a new user called 'koha' and give it administrative privileges. You also must have access to the administrative, or the first user's password. Also, in Appendix E there is sheet on which all of the necessary passwords can be written for ease of recall.

the password that you set for the 'koha' user as it will be needed throughout the tutorial and will be called the 'administrative password'. For more information on downloading and installing Ubuntu, please refer to the Ubuntu community website at www.ubuntu.com/community. If you are unfamiliar with Ubuntu there will be a few tips throughout to help get you on the right track.

Adding the Linux command-line, or 'terminal' as it is called in Ubuntu, to the task-bar will be useful for quick access throughout the tutorial and your life as a Koha administrator. To put the terminal on the task-bar, start by going to the upper left hand corner and click on the dark box with the Ubuntu symbol on it. This brings up a few of the most frequently used programs on Ubuntu but it is also has a search box and we want to search for 'terminal'. This search area can be very useful for finding files and programs. After you type in 'terminal' it will bring up a few options and the one that you want will be a black icon with a greater-than symbol and an underscore. Drag and drop this box to the task-bar and it will affix itself there until you ask it to disappear. You can make it disappear by right clicking on the icon and, in the resultant drop box, click on 'keep in launcher'. It is important to remember that with the power of the command line you can install thousands of applications (called packages) but also delete everything on every hard drive attached to the computer: so be careful.

This tutorial is lengthy, but once you have gone through the installation more than once you will need only little help to get you through. There is a very short version of this tutorial with only very basic commands in Appendix D.

A convention of this tutorial is to put all code lines to be entered into the terminal in *Courier New*. This way they will be distinguishable from other texts and more obvious what should be entered. These lines of code are also almost always isolated as their own paragraphs for easy and unmistakable reading.

This tip will make more sense as you move long in the installation as there are many terms not yet mentioned. But remember to check this paragraph if you are having an issue with the installation process. If you run into major trouble, your instinct might be to reinstall Ubuntu and start from scratch. While this is a great way to become comfortable installing something because you will be repeating the process, the better solution might be to delete certain files and to start again. Here are the two best examples:

- If you have completed the Koha installation and have logged into Koha and make a large mistake so that the installation is not longer usable. You could start by deleting the database 'koha' using phpMyAdmin first. You will be required to recreate the database in MySQL, grant privileges to the two users and go through the web installer again but it could take care of a deletion error or a database issue.
- If you have gone through the 'actual installation' section and have decided that things are not working correctly, you could always delete the 'koha' folder and reinstall it using Makefile.PL. Do not delete the koha-clone folder which is the one that you downloaded but the folder that is created as you go through the Makefile.PL section below. Deleting the 'koha' folder essentially rolls back the Koha installation to the point where all prerequisites have been installed.

Root Access

We will now do something that many books will not tell you which is how to log into the computer as the 'root' user. The root user is the most administrative user on the computer; the root account has access to every folder, can install and delete any files and can do a large amount of damage as well. However, this tutorial assumes that you will go through the entire process as the root user but once you have completed the full installation process, be sure to do as much as you can as a regular administrative user. When we install Ubuntu, it automatically assigns a password for the 'root' account so that processes can be run in the background but again we will change this password so we can log in as the root user ourselves. The best example of what you can only do as the root user is to access the web-server's configuration files (which we will do later). If you try to open them (after we install the program) as a regular user, the system will deny you access. To gain access to the root account, open the terminal and it will display the following:

To run a command as administrator (user "root"), use "sudo <command>".

See "man sudo_root" for details.

```
'username'@'computer-name':~$
```

The bold portion on terminal is called the prompt and it is used to enter commands. Commands are strings of numbers and letters which essentially amount to things that we ask the computer to do.¹¹ It will appear without the quotes; my prompt happens to be 'koha@koha-desktop:~\$' as I am logged in as the user 'koha' and I named the computer 'koha-desktop' during the installation process. This prompt can be very useful as it can tell you where you are in the file system and as what computer/user you are logged. Most importantly the prompt is used to enter command. For example, we want the computer to allow us access to the root user. At the moment, the only thing stopping us is that the root user has a password that we do not know. We must now ask the computer to allow us to change that password. To start the process, enter the following code in the terminal or command line:

```
sudo passwd root
```

What this line essentially says is "I have administrative powers" (the 'sudo' part) and "I want to change a password" (passwd) and "the name of the account whose password I would like to change is 'root'". When you enter the line above and press enter you will be asked to enter your administrative password. After you enter the administrative password and press enter the terminal will print: "Enter new UNIX password:". At this point you can change the root password to whatever you would like. I recommend, if you are not using this as a production system, that you use the same password for everything at first as there will be many passwords to remember or at least use the sheet from Appendix E to keep all of the passwords safe and accessible. It will ask you to enter the new root password again and, if the passwords match, it should return "passwd: password updated successfully" and then return

¹¹ An example of a command would be to type 'exit' into the terminal. It recognizes the word as a please-close-the-terminal command and exits the program.

to the default prompt. Now you can log into the system as the root user. To do so, start by logging out of the current administrative user and, once the system has stopped at the login screen you will see the user name you were just using as well as 'other' and 'guest user'. Click on 'other' and enter 'root' as the username and press enter. It will then ask for a password for the 'root' user which you just changed; so enter the new password you just created. Now, the computer will log you in as the root user and you will have a much better capacity to perform functions.

Note: The root user is an entirely new/different user than the regular administrative user so be sure to change any user settings you would like but be sure to move the terminal to the taskbar as we did before. Now we are ready to move on to prerequisites.

Packages

A quick note about Ubuntu and its differences from Windows. Instead of programs, any piece of software that is loaded onto your Ubuntu system is called a 'package'. There are also a few package managers that are very helpful with installing and removing packages as they contain search functionality and categorical searching. There is one package manager that comes with the current, as of late fall 2011, version of Ubuntu (11.10) which is called Ubuntu Software Center. It is a very user friendly application and can be used to install all of the LAMP prerequisites for Koha. You can also use the more robust Synaptic Package Manager which allows access to more packages but is a little harder to use. Again, both have search functionality and are very useful and will need to be used for the next section which is installing the -AMP in LAMP. However, as Synaptic does not come installed on the system you will need to install it using Ubuntu Software Center.

LAMP (Linux, Apache, MySQL and PHP)

Overview

LAMP stands for Linux, Apache2, MySQL and PHP (or Perl). These programs are an open-source software stack that can be used to create a full service web-server for home or enterprise functions. The power of this software stack is enormous as it is the software that Go Daddy, the large website hosting company, uses this to host its millions of websites.

L- is the operating system Linux which you already have installed on your computer in the form of Ubuntu . An Operating System (OS) help programs like Apache, or something as simple as the calculator, interact with the computer hardware to perform functions. Linux has been around since the early nineties and has been a great operating system for desktops and servers seeking a stable platform.

-A- stands for Apache2 which is one of the best known and definitely the most used web server in the world.¹² Apache receives requests for web pages from the outside world and sends out the

¹² Coar & Bowen, 2008

correct pages depending on the request. It may sound simple, but with Apache you can host hundreds of websites with hundreds of pages in each and, if set up correctly, Apache will handle the task gracefully. It may sound daunting but there are thousands of modifications that can be done to increase Apache's functionality. But do not fear! We will only be using two of them and it is already built in when we install it.

-M- stands for MySQL which is an open-source database administration tool owned by Sun Microsystems. MySQL allows you to create databases and queries in not only an easy to use way with the addition of a graphical user interface module but is also very fast, reliable and can handle multiple requests at once!

-P stands for either PHP or Perl. PHP and Perl can be used as server-side scripting languages but Perl, the language in which Koha is written, has a bit more functionality than PHP. Most importantly, both of these languages can be used to take web requests and use them to pull data from MySQL databases.

Do not worry about PHP or Perl as they are already installed on the system or will be installed when you install Apache2. Nor do we need to worry about the L as Linux you have, theoretically, already installed the Linux operating system as a prerequisite to going through this tutorial.

Now, to actually install the -AMP portion you have a choice between two installation options: using a package manager or automating the installation using a program called Taskel. Using a package manager is a manual process but the process of learning to use a manager is extremely useful because it gives you the ability to install any package available. I have detailed how to install the -AMP portion using both the manual package manager way and the automatic way using Taskel. However, it would be a good idea to take the manual track the first time through if you are not familiar with package managers.

As mentioned before, there are two different package managers to choose from: Synaptic Package Manager and Ubuntu Software Center. The following walk-through assumes that you are using Synaptic Package Manager. However, the new version of Ubuntu requires you install Synaptic using Ubuntu Software center as it does not come pre-installed. If you find Synaptic too complicated or if you just find Ubuntu Software Center more appealing, you can do the following walk-through with Ubuntu Software Center and all you will need to do is use the same search terms for the Synaptic version below and put the terms in the Ubuntu Software center search. The interface will be different but all that you will need to do is click on the package name and click the 'install' button.

LAMP Installation (Manual and Automatic)

Manual Instructions

To start, we need to install Synaptic Package Manager using Ubuntu Software Center. Open Ubuntu Software Center and in the upper right hand corner there is a search-bar. Type in 'Synaptic Package Manager' and click search. Synaptic should be the first on the list and you will need to click on the name. When you click on the name, an 'install' button will appear on the right. Click 'install' and it will install itself.

Be sure to add Synaptic to the taskbar on the left for easy access; all you need to do is to search for Synaptic using the search function used to find the terminal. Once Synaptic is open, click on the search bar at the top, enter 'apache2' and hit enter. There are many options available but the best packages are the ones with the little red Ubuntu symbol next to them. Click on the check box next to the package named only 'apache2'. Not 'apache2-doc' or any of the ones with '-mpm-' or '-mod-'. A small drop box will appear with options; click on 'Mark for Installation'. Once you click 'Mark for Installation', a window will appear asking if you want to mark all of the packages related to Apache2; click 'Mark' because all of the related packages are crucial to the running of each of the programs. You will now see many of the other packages on the page have checked checkboxes and are highlighted. These are the related packages that will be installed. By checking the box and marking the packages for installation, you are putting the package in a holding area; we will select our other package and place it, too, in the holding area and then we will ask Synaptic to install both packages at once. Next, search for 'mysql-server'. There will be a few options and you will want to install the latest version so go ahead and click on the box again and mark 'mysql-server-5.1' for installation and say 'Mark' to any related packages.

We will now ask the package manager to install the packages. Click 'Apply' in the upper left portion of the window and it will go through the installation process. Synaptic will start by giving you a summary of what is to be installed. Click 'apply' to move on. It will then show a window of the download progress of the packages including how many packages are left to be downloaded as well as how much time is left.

Once downloading is completed you will be asked a question. It will ask you to set a password for the 'root' use of MySQL. Enter a password but it would be a good idea to make it the same password as the root password we set up before. Hereafter, this password will be referred to as the 'MySQL root password'. It will ask you to enter the password twice for security purposes but after that is entered and verified both Apache2 and MySQL will be installed on your computer.

Automated Instructions Using Tasksel

The Ubuntu Community website calls Tasksel “[an] Ubuntu tool that installs multiple related packages as a co-ordinated 'task' onto your system”.¹³ We are going to install Tasksel and the –AMP in just that in two quick steps. First, we must install Tasksel and we do that by opening the Terminal (though you could install this using a package manager) and entering the following code to install:

13 Canonical Ltd., 2011, Tasksel

```
apt-get install tasksel
```

Taskel will then print out some code about accessing package repositories and such but then will ask you if it is okay that Taskel takes up about ~10MB on your hard drive. Hit the 'y' key for yes and then press enter. It will now spit out a bunch of lines of code as it downloads and then installs Taskel. Once it reverts back to the prompt you will be able to install -AMP with the following lines of code:

```
tasksel install lamp-server
```

It will then switch to a graphic screen where it will say it is downloading twenty-three packages and it will stop after a little while.

You will be required to create a password for the MySQL 'root' user and enter it again for verification. From now on this password will be called the 'MySQL root password'. Again, it is a good idea to keep all of these passwords the same for the sake of simplicity. After installation, the screen will revert back to the basic prompt and then you will be able to test the installations (instructions are below).

Extra Packages for MySQL

For ease of using MySQL, we need to install two more packages. They are called phpMyAdmin and MySQL Administrator. Both applications are designed so that an administrator is not required to use the command line to perform some of the most basic functions. It should be noted that all of the functions of these two programs *can* be complete with the command line. However, the same can be said for Windows or Ubuntu but having the graphical interface option will make things easier (especially for beginners):

- phpMyAdmin is a third party tool that allows visual access to your MySQL databases. It is very easy to use once learned and will help you to test and formulate queries, set up databases, insert rows, columns tables all with the ease of an intuitive graphical user interface. phpMyAdmin is not taught in this tutorial but there are many tutorials online and the creators of the software have created a \$2 book that walks you through its basic use.¹⁴ There are many other options for learning how to use this tool and links will be available when we log into phpMyAdmin the first time. We do not actually use phpMyAdmin in this tutorial but in many situations in a Koha library familiarity with this tool will help you gather data on your library's holdings, use statistics and general bibliographic data. For example, if you wanted to gather all of the ISBNs for all of the books that you have in your library, you could easily do this with one query, export it to a text file and then bring the ISBNs into excel for perusal. It is a wonderful and important tool with which to be familiar.
- MySQL Administrator is not only the administration tool that is created and recommended by Sun Microsystems (owners of MySQL) but it also provides a wealth of information about your MySQL server and allows you to change many of MySQL's settings. It can:

¹⁴ phpMyAdmin devel team, 2011

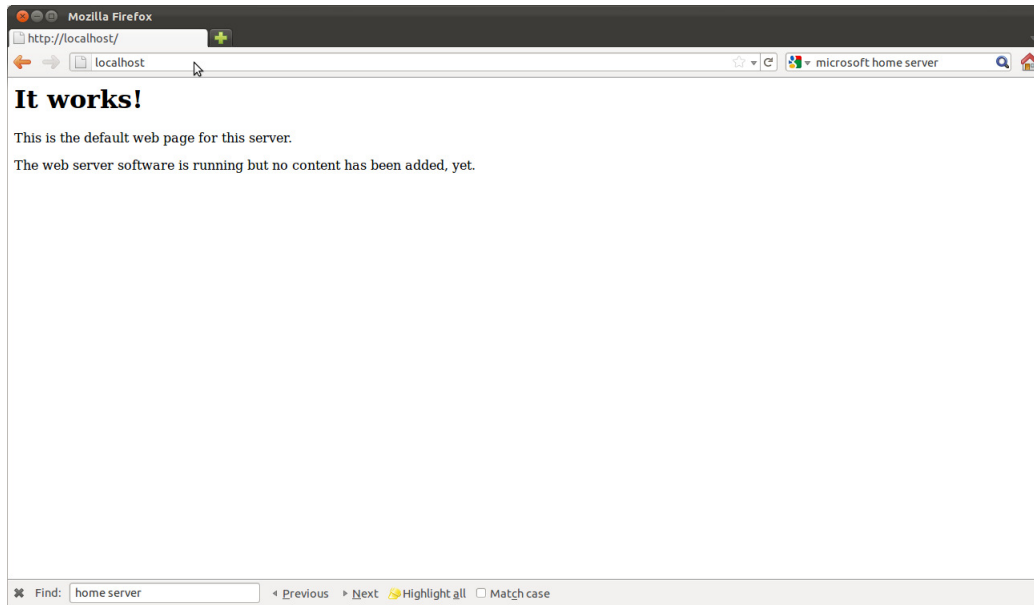
- Help you backup/restore your databases
- Organize your logs
- Monitor Usage
- Administer Users and User Settings
- Start/Restart the server

For the purposes of this tutorial you will only use this tool to add users but it is a good idea to learn about the tool as it will help you become a better MySQL administrator. More information about the use of MySQL Administrator can be found on the MySQL Administrator web page: <http://dev.mysql.com/doc/administrator/en/>

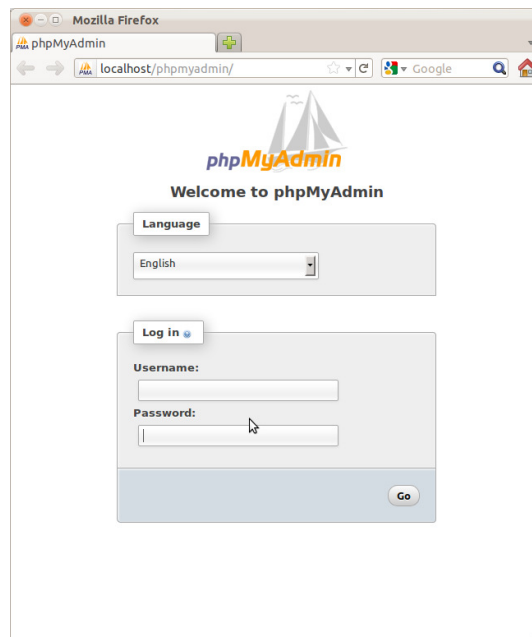
Search for both of the packages in Synaptic (or Ubuntu Software Center) by name ('phpmyadmin' and 'mysql-admin'), mark them for installation (both have related packages) and click Apply. phpMyAdmin will ask you a few questions. The first will ask you to which of the web servers on your system you would like phpMyAdmin attached. There will be two: apache2 (which you just installed) and lighttpd (which is a low functionality web-server that Koha does not use). Be sure that apache2 is checked and click 'Forward'. Next, it will ask you a question that you need only make sure it is checked; do not worry about this question! Just check the box and click forward! The important question is that it will ask you for a MySQL root/administrative password. Enter your MySQL root password and click forward. Now you will create a password to log into phpMyAdmin. Create one, but it is a good idea to keep them all the same for simplicity, and reenter it when asked clicking 'Forward' as you go. This password will be called the phpMyAdmin password. After creating a phpMyAdmin password, we are ready to check the installations!

Testing the -AMP Installation

Testing Apache2, and phpMyAdmin are very easy to do but to test them extensively is a little more complicated but not required. To test apache2, open a web browser. Enter 'localhost' into the web address bar. What you should see at this point is a large "It works!" with some more information should appear like this:



Next, type in 'localhost/phpmyadmin'. This should take you to a login screen. That looks like the picture below this paragraph. Enter 'root' as the user name and the phpMyAdmin password and pressing enter should log you in. If you look at the lower right hand portion of phpMyAdmin there will be several links in the 'phpMyAdmin' section; these links lead to the aforementioned tutorials on phpMyAdmin that you might find useful in study of the software.



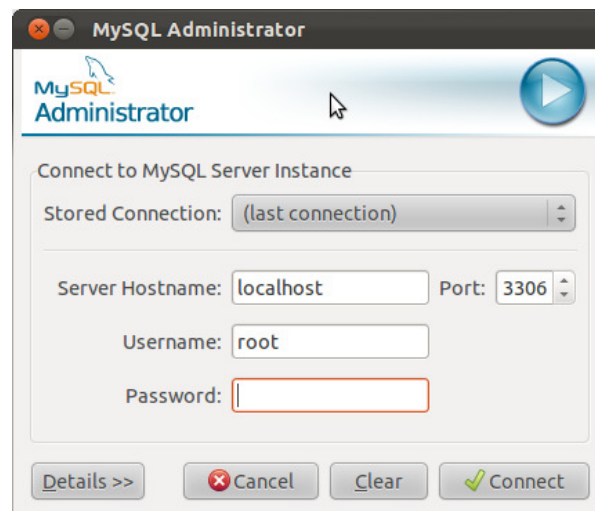
If you have successfully logged into phpMyAdmin and saw the “It works!” then both PHPmyAdmin and Apache2 have been installed successfully! If you would like to test the installation of your LAMP server more thoroughly the procedure is located in Appendix B. This Appendix, while testing the installation more thoroughly, explains how to simply display database info in web pages.

Note: Localhost is a great invention that allows a user to enter ‘localhost’ into a web browser and essentially what happens is the request loops back to the computer itself instead of going out into the world. So when you type ‘localhost’ into the web browser you are asking the browser to read your computer’s own web files.

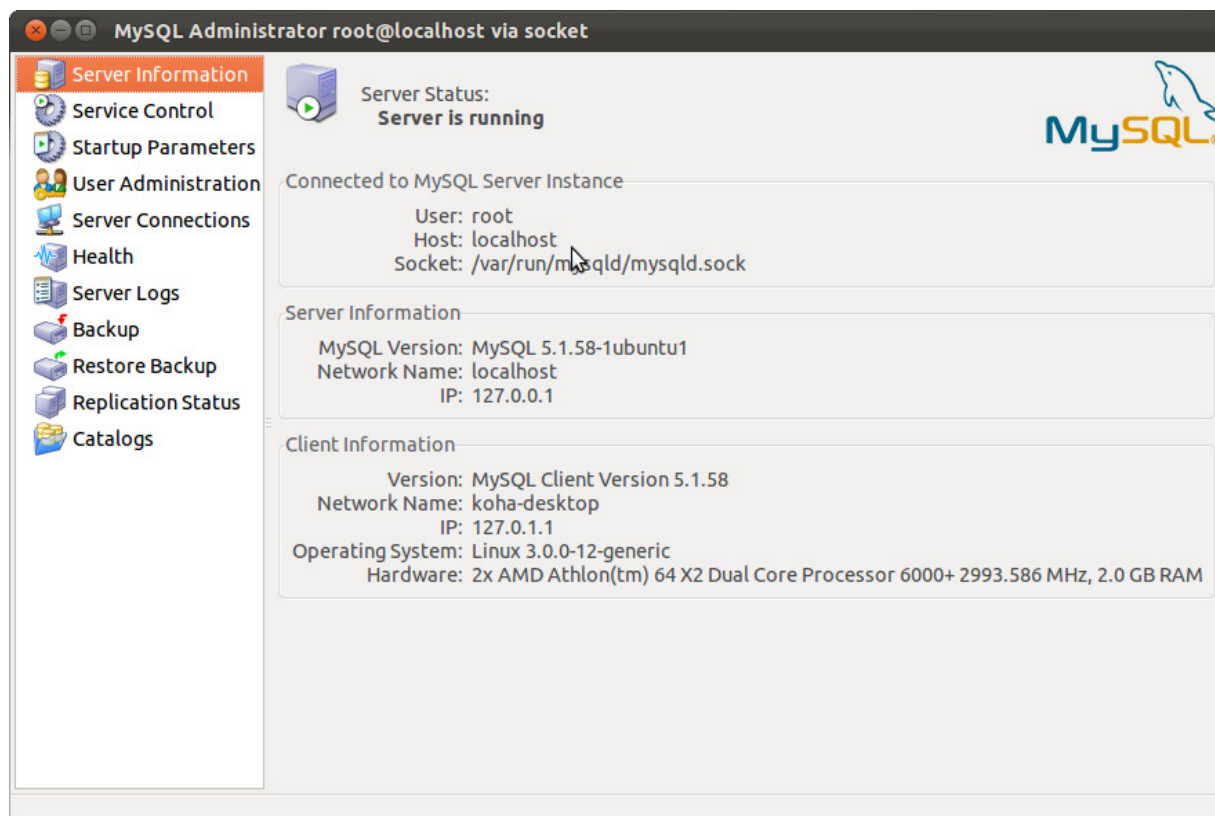
MySQL Setup

To setup MySQL for Koha, we need to add two users to MySQL, create a database called 'koha' and allow the new users access to the 'koha' database.

Remember when we installed MySQL Administrator when we installed mysql-server? We are going to use the MySQL Admin package for the first time. Start by clicking on the Ubuntu 'start' button in the upper left hand corner and do a search for 'mysql'. There will be three options: MySQL Administrator, MySQL Query Browser and phpMyAdmin. Click on or drag MySQL Administrator to the taskbar. Open the program and it will give you a login screen:



For 'Server Hostname' enter localhost because the database is on your computer; enter 'root' for the username and the MySQL root password. Click 'connect' after you have entered all of your information. The tool's setup will have options on the left and information about the highlighted or selected option on the right showing here:



Click on “User Administration” in the left toolbar and then on “+New User” which will appear on the bottom of the screen. The only things to worry about on this page are MySQL User and the New Password textboxes. Enter 'kohaadmin' into the user name box and 'katikoan' into the password box and re-enter into the confirm password box. 'kohaadmin' and 'katikoan' are the canonical user name and password and should be used at first as they are referenced often. Now click 'Apply Changes' and you have added the user! You will also need to add another user called 'kohauser' with a password of 'zebrastripes'.

Now we will create the database and the users, we need to give permissions to the kohaadmin and the kohauser.¹⁵ Start by opening the Terminal. You are going to log into MySQL through the Terminal and the formula for doing so is to call MySQL and to type '-u' and then the username to be logged in as and then '-p'. After typing this and pressing enter it will ask you for the password for the user you have entered. Here is an example of someone trying to log in as the user 'root':

```
mysql -u root -p
```

¹⁵ Permission issues are precisely the reason why this tutorial teaches you to access the system as a root user. The problem with being a regular user (and the reason why we must grant permission to the koha user) is Ubuntu will block access to anyone that does not have permission to do something. For example, if the koha user tries to install a package, Ubuntu will always ask for an administrative password because koha does not have permission to install anything. Permission must be proved by entering in the administrative password. All of these issues that could arise are removed on account of using the root user for everything during this installation.

Go ahead and type out `'mysql -u root -p'` and press enter. It will now ask you for the password for the user you have entered and in this case you will need your MySQL root password. After entering this and pressing enter, the terminal will print out about ten lines of information you should read. At the end there will be a default prompt for MySQL that looks like this: `"mysql>"`. To create a database we use the following command:

```
create database koha;
```

You must include the colon or MySQL will not execute the command. If you forget the colon it will accept the command but will return exactly this:

```
->
```

(Literally, space space dash greater-than) When it returns the above you will now be able to enter the colon and press enter at which point the command will be executed. When you successfully enter the create database command above, it will return 'Query OK, 1 row affected'.

Now we will grant privileges to the user kohaadmin. The formula is `grant all on <database>.* to '<username>'@<server>' identified by '<password>';` This is what you should enter but it is also a good example of granting all privileges to a user:

```
grant all on koha.* to 'kohaadmin'@'localhost' identified by 'katikoan';
```

Now grant all privileges to the kohaadmin and kohauser users using the same method. After granting permissions to both users we need to reset the MySQL privileges by entering:

```
flush privileges;
```

At this point the database is created, the users have all of the permissions so we have completed the MySQL setup. Now enter the word 'exit' into the mysql prompt and it will take you back to the regular terminal prompt. Now we must download Koha.

Where and How to Download Koha

Koha can be downloaded from a few places but we will be downloading it from the Koha Community server. Start by opening the terminal. At this point, if you do not know very much about navigating the Linux file system either using Nautilus (Ubuntu's graphical file navigation package) or the terminal, there is a small tutorial in Appendix A. If you are logged in as the 'Koha' user, you can stay in the current directory but if you are logged in as the 'root' user you will need to navigate to the 'koha' user home folder. If you are the root user, start by entering the following line:

```
cd /
```

Literally, Change Directory to `'/'`. This will take you back to the root directory of the hard drive and from there you will want to navigate to the 'koha' user's home directory by entering the following:

```
cd home/koha
```

Now we need to install the program that will help us to download the Koha installation files. It is called 'git' and it is very easy to install. Enter the following code:

```
apt-get install git
```

The terminal will start by spitting out a few lines of text and then it will ask you if it is alright if X number of megabytes are added to the hard drive. Say yes by typing 'y' and pressing enter. A few more lines will be spit out and it will be installed in less than a minute.

At this point you are ready to download the installation files. This will take quite a while but it is automatic and you will not need to sit at the computer and watch. When you enter the next line of code, an exact copy of a folder that is on the Koha-Community website (really the server) will be transferred to the current terminal directory, which, in this case, should be the home folder of the user 'koha'. Type in the following code to download the folder from the Koha-Community server:

```
git clone git://git.koha-community.org/koha.git
```

This could take up to an hour to download depending on your internet connection and the activity on the Koha-Community server. After it is completed it will revert back to the regular prompt. Now type 'dir' into the terminal which will show all of the files in the current folder. You should see a folder named 'koha' and that that folder contains the installation files.

After the download is completed, you will want to change the name of the folder that you just downloaded from 'koha' to 'koha-clone'. Perl Modules are the last bit to install before the real meat of the installation.

Perl Modules

Perl Module Overview

Perl Modules are small programs that are installed onto the computer which are required by Koha to run. A great understanding of what Perl Modules are is not required to install Koha so a larger explanation will be left out for the sake of simplicity. For information on Perl modules please visit the CPAN website at www.cpan.org. Perl modules are installed in two different ways but it is easiest to install them automatically. In fact, most of the modules needed for this Koha installation can be done so automatically but there are seven Perl modules that must be installed manually using the world-wide Perl repository called CPAN. We will start with the automatic installation first.

Module Installation

Automatic Perl Modules Installation

To automatically install most of the 200+ needed modules, start by opening the command line window or 'terminal'. You will need to install a program that will help allow you to automatically download the needed modules and install them for you. The program is called 'dselect'. Essentially, what we are going to do in the following process is to tell the computer to read a text document with the list of modules and tell it we want them installed using dpkg.¹⁶ You will then tell dselect to download, install, configure and remove any unnecessary installation files. To begin, you will need to install dselect by entering the following code:

```
apt-get install dselect
```

The window will fill up with the computer returning text as it installs dselect but it will not take more than a minute to install the software. However, along the way it will say 'setting up dselect' and then return to the normal command prompt. At which point the dselect installation will be complete.

We must now ask dpkg to read the list of necessary packages for Koha. The list is located in the Koha folder that we have previously downloaded.

```
dpkg --set-selections < /home/koha/koha-clone/install_misc/ubuntu.packages
```

Nothing will seem to occur as the system will not return any text. Behind the scenes, dpkg read the text file and it will finish after a few seconds and return to the normal prompt. Now we are ready to ask dselect to install all of the packages that dpkg just read. Enter the following code into the command line:

```
sudo dselect
```

At this point it will take you to a menu screen in the terminal which will give you a list of options. There are a few others but the only ones that we must worry about are:

- Install
- Configure
- Remove
- Quit

And that is the order in which you should run them. The system will remember where it is in the process and will move the cursor on to the next step so really all that needs to be done is for you to wait and to press enter. However, it is best to make sure that they are done in the proper order and that you do not close your eyes and press enter! Be sure to run the install command first which will then the print out screen after screen of information and for quite a while. It will also ask you if it is alright to use X amount of hard drive space. Say yes by typing 'y' and pressing enter. It will then print many lines and appear to be downloading items which it will do for quite a while. A screen will come up that will ask

16 Software in the Public Interest, Inc., 2011

you how you would like your mail server set up. Just press enter on the first and second screen as this will not matter to your Koha installation. Finally, it will ask you if you would like to erase all of downloaded installation files called '.deb files'. Either answer is fine so say yes or no and press enter. It will then say 'Press enter to continue'; go ahead and press enter again.

After it erases or ignores the installation files depending on your response, it will take you back to the menu screen and the highlighting will be over the next step. In this case the next step will be to configure all of the packages that were just installed. Make 100% certain that the highlighting is over number 4 or the command called 'Config'. The configuration process will take seconds at longest. The next steps are to Remove and then Quit; both of which will take seconds as well. At this point you are done with installing the large number of Perl modules that can be done automatically.

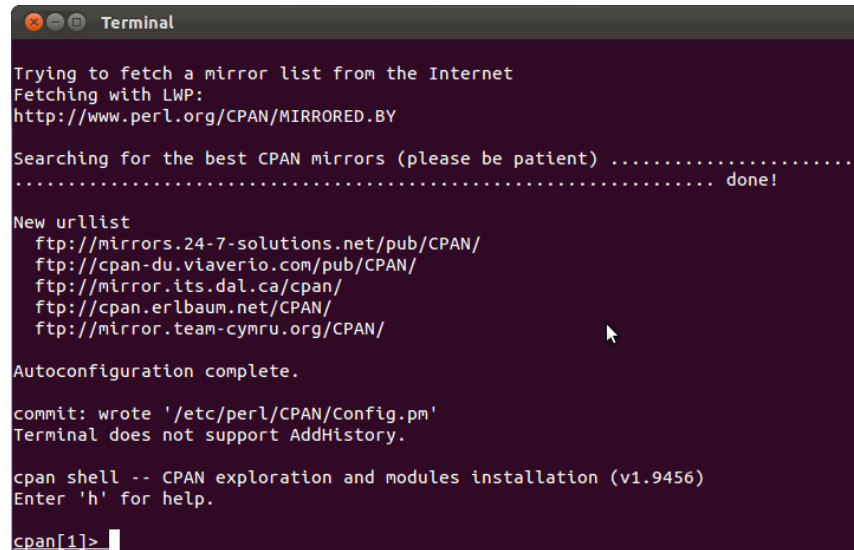
Manual Perl Modules Installation

The next step is to install the seven Perl modules that cannot be installed automatically using CPAN. CPAN is the largest repository of Perl modules and is the place from which we will install the other seven modules. These Perl modules are not sensitive as to where the terminal is in the file system so do not worry about changing the directory. Start by entering this into the terminal:

```
cpan
```

CPAN must first configure itself on your computer and then you will be able to install modules. It will first ask you if you would like it to configure as much of itself as possible with standard settings. The '[yes]' at the end of the line means that 'yes' is the default answer and all that is required to respond with 'yes' (the default answer) is to press enter.¹⁷ But again, here we want to say 'yes' so go ahead and press enter. Next, it will ask you if you would like it choose mirrors for you. Press enter so that it will configure the mirrors for you automatically. It will then ask many questions to the tune of "Would you like CPAN to prepend the installation?" Just hit 'enter' every time as the configuration will take care of itself. At the end you will see this:

¹⁷ In order to change the default answer you must type a proper response such as 'no' and press enter.



```

Terminal

Trying to fetch a mirror list from the Internet
Fetching with LWP:
http://www.perl.org/CPAN/MIRRORED.BY

Searching for the best CPAN mirrors (please be patient) .....
..... done!

New urllist
ftp://mirrors.24-7-solutions.net/pub/CPAN/
ftp://cpan-du.viaverio.com/pub/CPAN/
ftp://mirror.its.dal.ca/cpan/
ftp://cpan.erlbaum.net/CPAN/
ftp://mirror.team-cymru.org/CPAN/

Autoconfiguration complete.

commit: wrote '/etc/perl/CPAN/Config.pm'
Terminal does not support AddHistory.

cpan shell -- CPAN exploration and modules installation (v1.9456)
Enter 'h' for help.

cpan[1]>

```

When you see this screen, CPAN will have finished configuring itself and you will be ready to install the modules we need. To install modules we must ask CPAN in a formulaic way. You must first tell it that you want to install something and then on the same line you must put the name of the module. The first module is called 'Gravatar::URL'. The name of the module is both words with the colons so the colons must be present in order for the cpan to install the module. Be sure to copy the line exactly as CPAN is case sensitive. To install Gravatar::URL we enter:

```
install Gravatar::URL
```

This module takes a little bit of time to install but does not have any questions you need to answer along the way. After you understand the concept of installing these modules it is easy to install the rest. Just type 'install' then the name of the module and press enter for **any** questions. To make sure that they have installed successfully, essentially you have to look along the last ten or so lines once an installation has stopped and returned to the 'cpan[#]>'. It will say something like "Result: PASS" or the like showing here:

```

root@koha-desktop: ~
/usr/bin/make -- OK
Running make test
PERL_DL_NONLAZY=1 /usr/bin/perl "-MExtUtils::Command::MM" "-e" "test_harness(0,
'inc', 'blib/lib', 'blib/arch')" t/01.t t/02_open.t t/pod.t
t/01.t ..... ok
t/02_open.t .. ok
t/pod.t ..... skipped: Test::Pod 1.00 required for testing POD
All tests successful.
Files=3, Tests=10, 1 wallclock secs ( 0.02 usr 0.00 sys + 0.36 cusr 0.03 csy
s = 0.41 CPU)
Result: PASS
  REDTREE/PDF-API2-Simple-1.1.4u.tar.gz
/usr/bin/make test -- OK
Running make install
Manifying blib/man3/PDF::API2::Simple.3pm
Installing /usr/local/share/perl/5.12.4/PDF/API2/Simple.pm
Installing /usr/local/man/man3/PDF::API2::Simple.3pm
Appending installation info to /usr/local/lib/perl/5.12.4/perllocal.pod
  REDTREE/PDF-API2-Simple-1.1.4u.tar.gz
/usr/bin/make install -- OK

cpan[5]>_

```

Enter the following lines of code subsequent to the completion of the last module installing.

```

install Locale::Currency::Format
install Memoize::Memcached
install PDF::API2::Simple
install Text::CSV::Encoded
install Business::ISBN
install CGI::Session::Driver::memcached
install HTTP::OAI

```

After these installations are completed you have technically installed all of the prerequisite modules and packages and you are ready to start the real meat of the installation. In order to exit CPAN type 'exit' into the normal CPAN prompt and it will take you back to the default prompt.

Koha Installation

- Actual Installation process

Actual Installation Process

In the terminal, navigate to the Koha folder you have previously downloaded so the prompt looks like this:

```
<username>@<computername>: /home/koha/koha-clone#
```

Once you are in this directory you will be entering three lines of code and the actual installation will be done. There will be a few more things to configure such as getting Apache2 to work with Koha and we will need to set up the database but these three lines are the real meat of the installation. Start by entering the following code:

```
perl Makefile.PL
```

Entering this line starts the process of answering questions that will create a customized installation configuration file. It will be customized as you answer in respects such as the type of installation, where the files will go, what user and password would you like, et al. Each question will have some information printed in the terminal about implications of your responses so be sure to read through them as well as the explanations below. Because there are so many questions they will be listed in order of appearance with explanations. The questions are numbered and each explanation is just below the number itself. The first question you will see is:

1. Installation mode (dev, single, standard) [standard]

If you remember from installing Perl modules you will remember that this first question has a default response [standard] but the other options are in the parentheses. All of the options can be used and are correct but for this example type 'dev' and press enter.

2. Configuration directory: [/root/koha-dev]

We need to change this directory to within the user 'koha's' home folder. Type `"/home/koha/koha"` and press enter. This essentially creates the folder where all of Koha's installation files will go. The files can be put anywhere but it can be important to leave all of the folders close together and which is in the Koha user's home folder.

3. DBMS to use (Pg, mysql) [mysql]

In this case the default is correct because we have been working with MySQL. DBMS stands for Database Management System and in this case we have been working with

MySQL so just press enter to allow the default. Pg stands for Postgre which is not a fully tested option for Koha yet.¹⁸

4. Database server [localhost]

Another default answer. The database server was installed on the current system you are using and we accessed it through the localhost. We could theoretically put the DBMS on a different server at a different location but that is complicated and not for this tutorial. Press enter and continue.

5. Please specify the port used to connect to the DMBS [3306]

Another default. Unless you plan on reworking the port system on your computer—which is not extremely difficult but, again, not for this tutorial—3306 is the answer for this question. Press enter and move on to question 6.

6. Please specify the name of the database to be used by Koha [koha]

This is an important question but just a few steps ago we created the database in the terminal and called it 'koha' so the default is correct. Press enter. For this question and the next two, you could theoretically answer whatever you would like as long as the name of the database you created for Koha and the answer to this question match. The same goes with the database user and password. But, in this case, it will be easier to just use the default values that given in this tutorial.

7. Please specify the user that owns the database to be used by Koha [kohaadmin]

The default is going to be correct for the rest of the questions but it is good to understand what you are agreeing to! 'kohaadmin' was the database user that we specified that will have complete control over the database.

8. Please specify the password of the user that owns the database to be used by Koha [katikoan]

Here is the same question as 7 but with password instead of username. Again, you could enter anything that you want but you would have to specify what the password is so that Koha will be able to access the database.

9. Install the Zebra configuration files? (no, yes) [yes]

Zebra is Koha's powerful search engine. It indexes the MARC records we load into Koha's database and returns them at incredible speeds. It is entirely free and it is very necessary if your library will have any number of titles over 10,000.¹⁹ If you would like to use Zebra, press enter to accept the default. However, this tutorial does not go over Zebra configuration.

10. MARC format for Zebra indexing (marc21, normarc, unimarc) [marc21]

¹⁸ Sirohi & Gupta, 2010 p. 20.

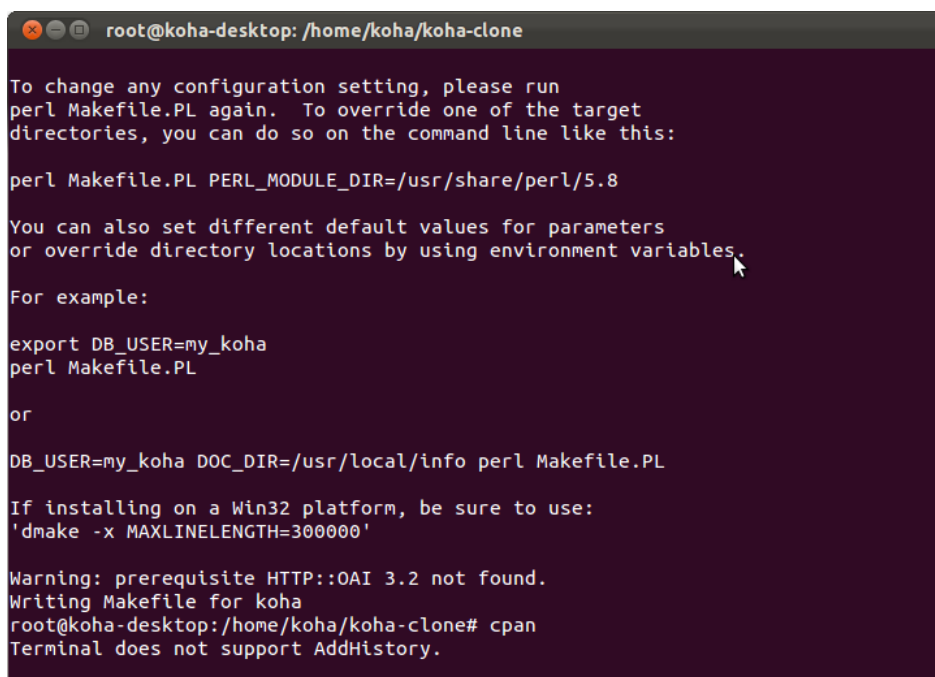
¹⁹ Sirohi & Gupta, 2010, p. 39

MARC21 is by far the most popular format at the moment. Most likely, especially if you are not familiar with MARC formats, you should choose the default. For information on MARC please see the Library of Congress' MARC page:
<http://www.loc.gov/marc/faq.html>.

Again, you should answer the rest of the question by okay-ing the default by pressing enter.

11. Primary language for Zebra indexing (en, fr, nb) [en]
12. Authorities indexing mode (dom, grs1) [dom]
13. Please specify Zebra database user [kohauser]
14. Please specify the Zebra database password [zebrastripes]
15. Install the SRU configuration files? (no, yes) [yes]
16. SRU Database host? [localhost]
17. SRU port for bibliographic data? [9998]
18. SRU port for authority data? [9999]
19. Install the PazPar2 configuration files? [no]
20. You will need a Memcached server running. (no, yes) [no]
21. Would you like to run the database-dependent test suite? (no, yes) [no]

Once you have given the default response to question 21, it will start printing out lines of code but will end at the default prompt. At this point you must look over the screen and read the last 25-30 lines in case there are any errors. The most important thing to look for is whether or not you have forgotten any prerequisites. The following screenshot has a forgotten Perl module:



```

root@koha-desktop: /home/koha/koha-clone

To change any configuration setting, please run
perl Makefile.PL again. To override one of the target
directories, you can do so on the command line like this:

perl Makefile.PL PERL_MODULE_DIR=/usr/share/perl/5.8

You can also set different default values for parameters
or override directory locations by using environment variables.
For example:

export DB_USER=my_koha
perl Makefile.PL

or

DB_USER=my_koha DOC_DIR=/usr/local/info perl Makefile.PL

If installing on a Win32 platform, be sure to use:
'dmake -x MAXLINELENGTH=300000'

Warning: prerequisite HTTP::OAI 3.2 not found.
Writing Makefile for koha
root@koha-desktop:/home/koha/koha-clone# cpan
Terminal does not support AddHistory.

```

Look in the same area where my forgotten module is located on your own screen for any missed modules. However, you should be alright if you installed all of the manual and automatic modules from the Perl Modules section above. At this point you are ready to enter the next line into the terminal:

```
make
```

Just as it is: just 'make'. This one will spit more lines at you than you care to handle. But it will take only a minute or two to run its course. This is the penultimate line:

```
make test
```

This one will spit out quite few lines but it is not nearly as dramatic as 'make' from before. There is not a need to look at any of the lines except for the last few printed before the default prompt reappears. It will give you the results of the test and it should say “Result: PASS”. If it says 'pass' continue with the last line:

```
sudo make install
```

This will take no time at all to return to the prompt but now go to the the 'koha' user's folder and look to see the new 'koha' file that you just installed!

Post Installation

Setting up Apache's Configuration Files (ports and http)

- Customizing the Configuration File

Web Database Installer

- Walk-through of Web Installer
- Testing and Login

“Next Steps”

- Using the installation

Setting up Apache's Configuration File

Customizing the Configuration File

At this point, Koha is installed on your system and now we need to set up the Koha-Apache relationship and allow them to talk to one another. We do this by changing Apache's main web-content configuration file. First, let's find the file Koha provides as a template for what we need to tell Apache.

Start by opening Nautilus and navigating to the Koha folder at `/home/koha/koha`. Next, enter the `'etc'` folder and open the file `'koha-httpd.conf'`. What you see now is a configuration file. All of the words and lines that you see are commands to Apache2 telling it do something. It is not necessary for you to understand more than 2% of what is going on here. But there are two important things to understand:

1. This document is filled with properties and values. Line 7 says `"ServerAdmin"` which is a property that Apache recognizes and then the email address that follows is the value. It is the same as saying `X=4`; `'x'` is the property and 4 is the value.
2. What this file is saying is that there are two websites being hosted on this one server. In order to do this we have to tell Apache what the names of the websites are and where the files are on the hard drive. The locations of the files on the hard drive are already there so the only thing that we have to do is tell it what the two websites are and how they are identified.

For now what we want to do is to copy (ctrl C) everything in the text document. Next, we want to navigate to the Apache2 folder which is located in `/etc/apache2`. What you will see in this folder are a few folders not to be worried about but also a few files. The first is `httpd.conf`. First, open it and paste all of the text that you took from the Koha version. This tutorial assumes that you will only be accessing Koha at the computer terminal through localhost. It is better to perform the installation as simply as possible the first time and one of the most effective ways of doing this is to use localhost.

There are a few things we need to change in order for this configuration file to work the way that we want it. Start by typing the following on the second line from the top: `"ServerName localhost"` (without the quotes). Be sure only to insert this line into the document and not to create any new lines by pressing the carriage return (or enter button) as we will be referring to line numbers in the text document.²⁰

Adding a server name was the easy part and now you will learn the 2% of what you need to understand in this file. Basically, this file says that there are two websites on this server. One is for the OPAC and the other is for the Administrative side of Koha here called the `'intranet'`. You will see on line 5 of the document (if you have not added any spaces when you added the `ServerName` line) that it says `'##OPAC'`.²¹ This is the start of the OPAC section. Then down on line 97 it says `'##Intranet'`: this is where the administrative portion starts. We need to change a few values so apache knows what we want.

We need to change the same things in both of these sections and the only lines that we need to worry about are the ones highlighted below:

²⁰ To add line numbers to the document be sure that the text editor is the window the active window and move your mouse to the gray/black bar at the top of the screen. There will be a familiar list of options on the left side of the bar and click on `'edit'` and then `'preferences'` at the bottom. One of the options will be to display line numbering. Check the check box and click close.

²¹ The hash mark or pound symbol means that this line is `'commented out'` and apache will ignore that line of the configuration document. You can basically `'comment out'` on anything in these lines as long as there are hash marks at the beginning. You can see that we can comment out properties and values but we can also use this to actually make comments such as the line about OPAC

```
<VirtualHost 127.0.1.1:80>
  ServerAdmin webmaster@koha-desktop
  DocumentRoot /home/koha/koha-clone/koha-tmpl22
  ServerName koha-desktop
# ServerAlias opac.mydomain.com
```

Start with the OPAC section at the top. Uncomment the ServerAlias and change the value from opac.mydomain.com to 'localhost'. Next, you need to change the ServerName as well as the VirtualHost to localhost. Leave the ':80' intact so the VirtualHost property's value is 'localhost:80'. Those five lines above should now look like this:

```
<VirtualHost localhost:80>
  ServerAdmin webmaster@koha-desktop
  DocumentRoot /home/koha/koha-clone/koha-tmpl
  ServerName localhost
# ServerAlias localhost
```

You will need to do the same thing to the intranet portion of the configuration file. The only difference between what we just did to the OPAC section and what we will do to the intranet section is to have the VirtualHost property say 'localhost:8080' instead of just 80. You also need to leave the ':8080' in the ServerName section of the intranet or administrative portion. So, essentially, the only difference between the OPAC and the administrative section is what we leave alone.

The last thing we need to do to this file before we are done is to uncomment line 3, NameVirtualHost, and put an ':80' after the asterisk. At this point you are done with this configuration file and can save and close out of it.

Next, we need to go back to Nautilus and open another file in the apache2 folder. It is called ports.conf. Open it up and add the following line:

```
Listen:8080
```

What you are essentially doing is asking Apache2 to listen for internet requests at port 8080 which is what the administrative side of Koha uses by default. After you enter this code save the file and exit. You are now done with *all* the configuration files but we need to do two more things before we can access the installation through the web browser.

First, we need to open the terminal and enter the following lines of code:

```
sudo a2enmod rewrite
```

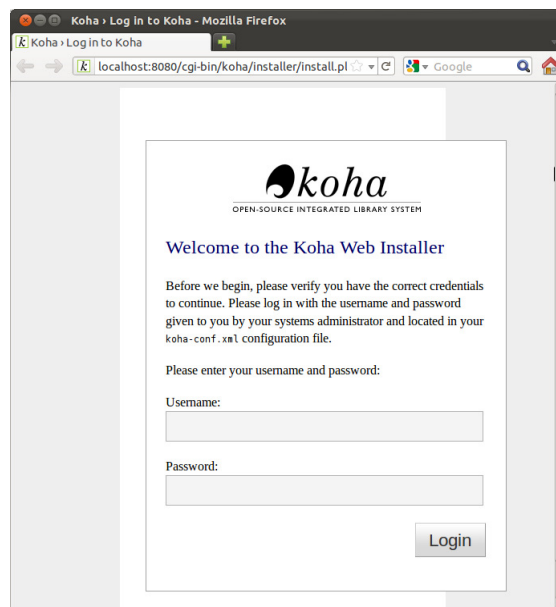
²² The Document Root property tells Apache where the web files are for the website

Essentially, what you are asking apache to do is to allow it to change URLs 'on the fly'. This portion is not something you must worry about but if you would like more information please see the apache2 documentation website.²³ Next enter:

```
service apache2 restart
```

We are asking apache to restart (as you could have divined). Restarting can not only fix problems but it requires apache to re-read the configuration file that we just changed and apply the changes. Thus, any time that you make changes to Apache's configuration or ports file, you need to restart Apache using the restart line above in order for the changes to take effect.

At this point you should be able to go to <http://localhost:8080> and you should see something like this:



All that you need to do is go through the database installer which is rather simple compared to what you have done so far!

Web Database Installer

Walk-through of Web Installer

²³ The Apache Software Foundation, 2011

Now you are ready to run through the web database installer and, as the name implies, it installs Koha's database tables. You will be going through an interface that is 5-6 pages in your web browser and, based on your responses, the installer will create all of the tables and columns and data that you need to start your way on using Koha! Open a web browser and go to localhost:8080. It will bring up a login screen and if you did not change the username/password from when we added usernames in MySQL Admin or when we installed Koha then the username should be 'kohaadmin' and the password should be 'katikoan'. On pressing enter, it will take you to your first (and easiest) question of what language you would like to use. The default is English but make sure it says 'en' in the drop box before you click 'next'.

When you clicked 'next', the installer checked to ensure that all of the prerequisites had been installed correctly. If everything checks out okay it will confirm it on this page. Click 'next' and move on.

The installer will now read your installation questions and confirm that everything is correct. Check over the few lines to ensure that they are correct. Here is what it should look like:



Click 'next' to move on.

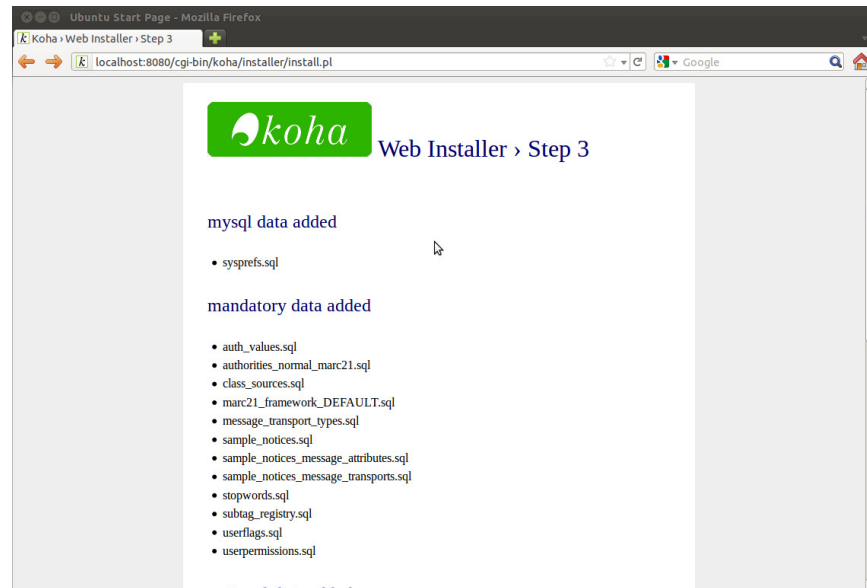
Now the installer will confirm the entries from the last page and should say that a connection was able to be established, the 'koha' database has been detected and the kohaadmin user has all required privileges. Click 'next' and it will take you to the screen which essentially says it is ready to install your first tables. When you click next on this page it will take a few seconds but when it is completed it should say "Database tables created" and a big "Success" at the top. There will also be yet another 'next' button.

The next screen says you will be doing some 'basic' configuration but you can do so much with next few screens to prep your installation for perusal and use. Click on the hyperlink to continue.

The next page asks which flavor of MARC records you will be using. MARC21 is generally more accepted at the moment and will therefore be more useful in finding copy-cataloging records. Click on the MARC21 radio button and click next.

This is the most important screen and requires careful consideration before continuing. There are two sections: one is the 'MARC Frameworks: MARC21' section and the other is the 'Other Data' section. Read through but make sure all of the boxes in the MARC Frameworks section are checked. Check all of the items in the second section as well except for the 'A few sample libraries...' and 'A few sample patrons....' Now click 'import' to import your settings. This import of all the things that you just checked will take quite a while but once it is done you will have completed the installation.

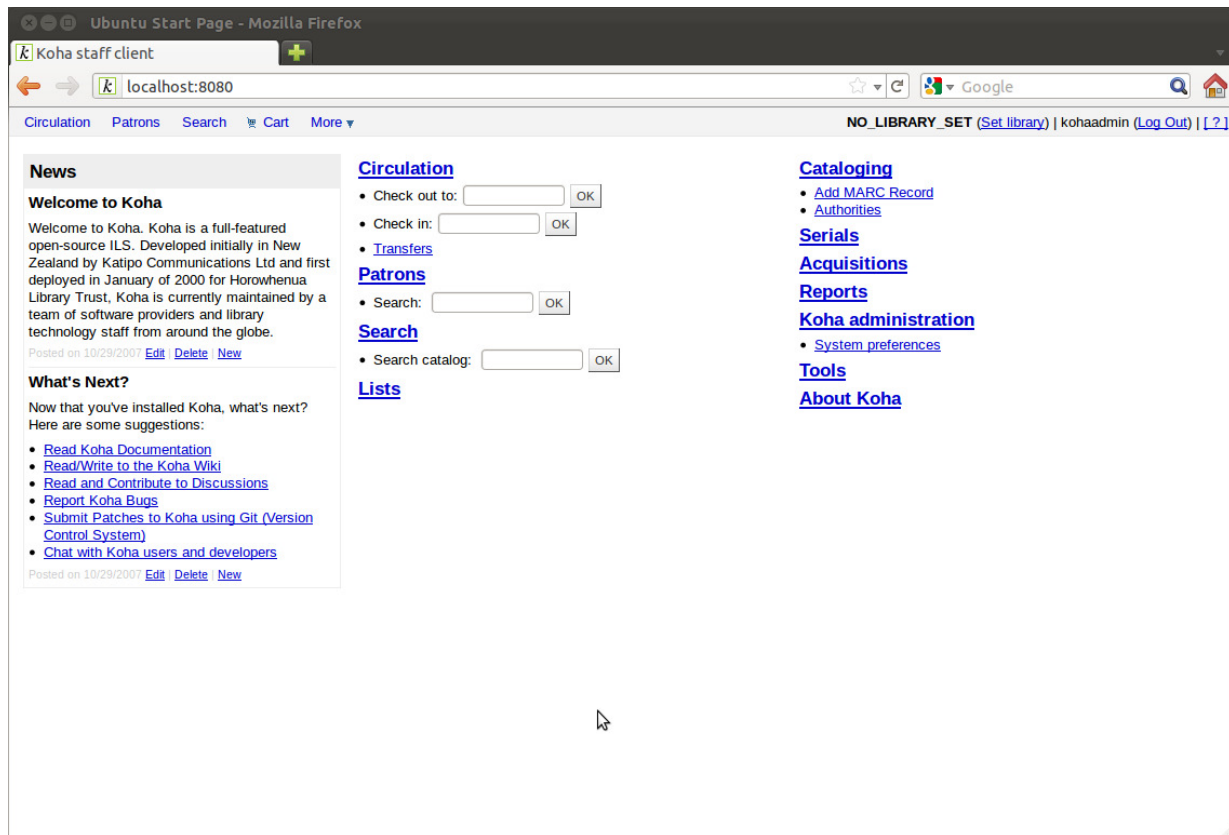
Once finished it will give you an import confirmation screen following:



This just shows you what was imported. Simply go to the bottom and click 'finish' and it will take you to the login screen for your new installation. The login will be localhost:8080.

Testing and Login

The login remains the same as whatever you chose for mysql admin but it will most likely be 'kohaadmin' and 'katikoan'. Go ahead and log in once the last installer screen redirects you. Once you have logged in it should look something like this:



You have completed the installation. Take a break and think about all that you have accomplished! (At this point the meat of the installation is theoretically completed)

“Next Steps”

- Using the installation

Using the Installation

You have installed Koha. So what now? It really depends on what you want to do with your installation. I recommend taking the approach of who you want to be in the library or the level of understanding you want to obtain:

- Circulation/copy-cataloging specialist: You would need to learn how to check books in and out as well as add patrons, do copy and original cataloging and take fines. There are plenty of websites as well as YouTube videos that can describe how these tasks are completed. After you import a few records into your Koha installation and create a few patrons, you could simulate checkouts.

- **Managing Librarian:** not only to learn all of the tasks of the Circulation/Cataloging specialist but also a host of administrative things such as setting up MARC frameworks as well as gathering use statistics. You would also need to learn how to set up budgets, funds and perhaps learn how to order things all in your Koha system. Taking this angle is more difficult than the Circulation route but would allow you to be fully prepared for running a library with a fresh Koha installation.
- **Database Administrator:** Here is where the phpMyAdmin comes in; in order to administer the database, you will not only need to know how to do everything in the library in terms of cataloging, circulation and administration but also where all of these items live in the database. In addition, you will also need to know about relational databases in general but also, and more importantly, about Koha's database and how it works. There are around 50 tables in the Koha database and each one performs a different function. A good example of a simple task that an administrator would face is how to find the average number of pages for all of the monographs in the OPAC. Luckily, there are many ways in which to learn about relational database including taking classes, learning online or buying yourself a good book.

Using these above examples of learning goals, you can decide what level of understanding of Koha you would like whether this is your first ILS or you are adding another to your repertoire. Again, there are many tutorials online, a few books and YouTube videos, but one of the best ways to proceed is to learn by doing. Start playing with the new software and when you run into issues search for the solution and, most importantly, do not get discouraged! Set goals and meet them!

My Thoughts on Koha

- Koha, Tablets and Netbooks
- Advantages
- Disadvantages

Koha, Tablets and Netbooks

Mobile library checkout stations have been talked about as a viable option not only to bring checkout to different areas of the library but also to expand the number of checkout stations during peak checkout hours. Two great options for mobile checkout stations using Koha are Netbooks and Tablets

Netbooks are a perfect option for mobile checkout stations with Koha as all that you really need is the Netbook itself, a connection to a wireless network and a barcode scanner. As Koha is a web based application, a netbook will be able to easily handle the small workload of Koha. They are also fairly easy to use as most netbooks come with Windows, which is preferred by most users. Staff will feel comfortable in the Net books/Windows environment though the screens are quite small.

Tablets are very close to being able to be used as mobile checkout stations for Koha though they may come with a few issues at the moment. One certain issue is that there are many tablets that do not

have full sized USB ports which is the main connective port used by barcode scanners in many of today's library checkout stations. Currently, there are only a few tablets that have full sized USB ports but it is becoming a great selling point which means the port will soon not be an issue as manufacturers will add them to stay competitive. The other issue that may arise is that of a driver for the barcode scanner. Drivers are computer programs that allow the computer to talk with a devices connected to the computer. For example, I am typing this tutorial on a keyboard and this keyboard is sending signals to the driver which turns the signals into something that the core of the computer understands. Currently, all tablets either run on a flavor of the Android operating system or Apple's IOS (eyeO.S.) neither of which are too focused on drivers for barcodes scanners. However, I only mention this because it is a possible issue and not because I have tested the theory and found it true.

You could choose any of the current tablets on the market that have full sized USB ports or there are attachments that create a USB port for the iPad that work quite well. A simple search for 'iPad barcode scanner' on YouTube will reveal many videos displaying how the iPad can be used with a barcode scanner. Tablets are usable now but the future is much brighter for tablets in the library world as Windows 8 is going to be a tablet friendly operating system and it will be picked up by tablet makers quickly.²⁴

Advantages

With Innovative Interface's Millennium, you must have a central server and then install software on each administrative computer in order to access the server. This requires either that you pay someone to install the software, pay and IT person or someone with a bit of tech knowledge install the software. At any rate, installing software consumes resources monetary and in terms of time. Koha requires a central server and from there can be accessed from any computer with a web browser making it cheaper for installations. Koha not being tied to specific hardware makes it platform independent.

This lack of platform dependency is advantageous in many ways. There are very divided operating system lines: most people like Windows but there is a veracious population that enjoy Apple's Mac OS. With Koha both parties can use their native operating system and can still access both the OPAC and the administrative side of Koha.

Koha has built, out of the box, in faceted searching. Libraries and their OPACs had fallen behind in faceted searching as it did not exist in many libraries' OPACs while large commercial websites had had the functionality for years. The Koha developers included faceted searching not only because it is very helpful to the patrons but because it is now expected as a helpful feature by patrons and staff alike.

Finally there is price. Theoretically, Koha costs nothing as you can download it at will and install it on your own systems. However, this is not advised for anyone but professionals. There is, therefore, the cost of support including hosting and data safety. To illustrate the difference in price between Koha (hosted through Liblime, a major Koha support vendor) and Millennium (provided by CW/MARS, a non-

²⁴ I do not have the best reference for you here but there is much buzz on the subject and if you look at the new Windows 8 operating system, you will see many features that lend themselves to tablets. There is a PC Magazine article here explaining the current thoughts on the Windows 8 tablets: Ziff Davis, Inc, 2011

profit providing ILS support and hosting for Central and Western Massachusetts libraries), I will show ILS service/hosting fees for a large Western Massachusetts Library. This is a real library with about 80,000 items in their catalog and currently a customer of CW/MARS. I received a quote for their sized library from Liblime and the cost for their hosting of Millennium was obtained directly from the director.

CW/MARS	
Initial Investment (est)	\$15,000.00
Year 1	\$22,000.00
Year 2	\$22,550.00
Year 3	\$23,113.75
Year 4	\$23,691.59
Year 5	\$24,283.88
Year 6	\$24,890.98
Year 7	\$25,513.26
Year 8	\$26,151.09
Year 9	\$26,804.86
Year 10	\$27,474.99
Ten Year Total	\$261,474.40

Liblime Koha Tier 2 50k - 100k records	
Initial Investment	\$15,000.00
Year 1	\$4,000.00
Year 2	\$4,100.00
Year 3	\$4,202.50
Year 4	\$4,307.56
Year 5	\$4,415.25
Year 6	\$4,525.63
Year 7	\$4,638.77
Year 8	\$4,754.74
Year 9	\$4,873.61
Year 10	\$4,995.45
Ten Year Total	\$59,813.53

Difference of: \$201,660.87

There is an initial investment cost as you need to catalog all of the titles/items in the library and the hosting vendor needs to set up their solution either in the library or on their own servers accordingly; the initial cost is at the top. The first year cost is listed below the initial cost but the second year and on each have an increase in cost at the rate of 2.5% over the previous year.²⁵ The yearly cost for Millennium is an amazing amount more than Liblime's Koha hosting. The difference in cost would cover a part time librarian, a few pages for shelving or perhaps, worst of all, around 5,700 books at \$35.⁰⁰ each.

One note about the cost for Millennium in this chart: CW/MARS also provide access to around 500 ebooks, a union catalog of all items across the state and a reporting system of users that are notorious at stealing books from the libraries. These features would add cost as you are not just getting a hosted instance of Millennium but you are buying into a consortium. The original amount that is listed in the library's line item for CW/MARS is \$28k-\$29k. I decided to reduce the full amount by 20% in order to get a more accurate picture of just the cost for hosting Millennium.

²⁵ The 2.5% increase was quoted by Liblime and is only available to customers who sign a five year service contract with them. The 2.5% increase for Millennium from CW/MARS is an estimated cost increase.

Disadvantages

Koha does have a few disadvantages as you are still attached to a vendor of some kind. Instead of paying CW/MARS you are paying Liblime or another vendor. It is still the same issue of not being able to host something in house because we are still dealing with people's information: phone, address, full name etc. This data needs to be protected and the best way to do that is to pay someone and blame them if anything goes awry.

Appendices

Appendix A: Nautilus and the Terminal: Navigating the Linux File System

Nautilus is the graphical folder navigation tool that comes with Ubuntu-desktop. Nautilus is almost exactly like Window's File Manager and looks and feels similar. We can use the terminal to navigate the file system as well but it is text based and for many users who never used the old DOS operating system it can be a daunting task to learn. We will first take a look at Nautilus and then terminal but we will then combine the two in an exercise to help explain how the terminal is not scary in terms of file navigation. One note before we begin is that this guide focuses on three folders in the Linux file system: the root folder, the home folder and the koha user's home folder. The most important thing to know is that the root folder contains the home folder and the home folder contains the koha user's folder.

Nautilus

Open Nautilus and look over the window and the layout. There will be a list of hard/CD drives, common folders and network drives on the left quarter of the screen or 'files section' of Nautilus. The other three quarters consists of the folders of the current directory on the bottom or 'folders section' with navigation tools at the top or 'navigation section'. On the right side of the navigation tools there is a search feature as well as arrows that are much like the back and forward buttons of any web browser. The left side of the navigation tools shows the folders that you have visited and their order from top to bottom. Top to bottom meaning the top folder contains the lower folder to the right and that right folder contains the folder to its right and on down the line.

Nautilus starts in the home directory of whatever the current logged in user is in this case it should be the user koha. The home directory is the same as My Documents in Windows which contains text documents, music, pictures and videos. To enter the pictures you click on the folder of the pictures. To go back one directory you click on the little silver arrow next to the home folder in the navigation section. You can also keep clicking that silver arrow to go back to the folder that contains the home folder and then continue clicking until you get to the root or top-most folder. A faster way to get to the top-most folder is to click on the 'File System' icon in the files section. To get back to the koha user's home folder from the root folder of the file system, click on the 'home' folder and you will see all of the home folders of all of the users and click on the 'koha' folder.

Terminal

Open the terminal and look at the prompt. As said before, the prompt will start in the home folder of the logged in user just as Nautilus. The prompt that comes up on my computer looks like this:

```
koha@koha-desktop:~$
```

It comes up like this because the user logged in is 'koha' and the computer name is 'koha-desktop'. Let's see what folders are currently in the koha user's home folder. Type 'dir' and press enter. This command lists all of the folders and files in the folder of the of the current terminal directory (the term directory and folder are interchangeable). You can tell in what directory the terminal is currently by looking at the prompt itself.

Here is how to tell the current directory of the terminal. It can be difficult to explain so I will do so through an example. Type 'cd /' and press enter; then type 'dir' to see what folders and files are in the current directory which is root. To get back to the home folder of koha we need to move from the root directory to the 'home' folder and then into the folder 'koha'. Type 'cd home' and the current directory will change from root to the home folder (not the koha user's home folder but the folder that contains all of the home directories). The prompt will change to:

```
koha@koha-desktop:/home$
```

As you can see, '/home' now shows that the current directory is now home. It is from this folder list area that you can tell in what directory the terminal is currently. Now type 'cd koha' and you will be back in the home folder and the prompt will look like this:

```
koha@koha-desktop:/home/koha$
```

These commands can be combined so that if you are in the root directory you can type 'cd home/koha' and you will move to koha's home folder.

Finally, the last command to learn is 'cd ..'. The double dot after a 'cd' takes the current terminal directory and moves it up one directory. So, if we are in the koha home directory and type 'cd ..' we will move up to the 'home' folder that contains all user's home folders. If we type 'cd ..' once more, we will suddenly be in the root folder. Here is a list of the change directory commands that we have gone over:

cd ..	Takes you one folder back
cd /	Takes you to the root folder
cd folder/folder	Takes you from the current folder
dir	Check the contents of the current directory

Terminal and Nautilus working Together

One great way to learn more about file navigation in the terminal is to play around with changing directories using 'cd' and following your movements in the terminal using Nautilus. Every time that you change directories in the terminal, do a 'dir' to see what is in the folder and make the same movement in Nautilus and check to ensure that the contents of the moved to folder are the same.

Appendix B: In Depth Testing of Your LAMP Server

This optional test MySQL and Apache2 requires a little phpMyAdmin knowledge but the best way to learn is to jump right in! Also, by learning how to do this, you will have learned how advanced websites work in serving information to browsers which can be very useful.

Warning!: Once you have updated the http.conf file in the Post Installation section, you will not be able to do this advanced testing. However, if you have gone through the web installer, again in the Post Installation section, the database will be fully tested and this in depth testing will, not be unnecessary, but .

In order to test MySQL more completely, you will need to go into phpMyAdmin and create a database. This database will only have one table with only one line in it but it is all that you will need to tests the LAMP installation. You will then create a php script (the script is below) that asks the database server to send us the information in that one line in that one table in that one database. To do so create a database called 'testing', a table inside that database called 'names' and then two columns called 'first' and 'last'. Insert one line into this table and have it be your first and last name in their respective columns or any name that you like. Next, open a text editor and past the following lines into it.

```
<?php

$host="localhost";

$user="root";

$password="DB PASSWORD";

$database="testing";


mysql_connect($host,$user,$password);

mysql_select_db($database);


$sql = "SELECT * FROM names";

$result=mysql_query($sql);

$num=mysql_num_rows($result);


while ($row = mysql_fetch_assoc($result))
{
echo ('<br/>'.$row['last'].', '.$row['first'].'<br/>');
```

}

?>

Remove 'DB PASSWORD' (leave the quotes) and enter your MySQL root password. Save the file to the desktop and call it 'juno.php'. Next, copy that file from the desktop to the place where Apache2 gets the webfiles to server to requestors which are located at /www/var. There should already be a file called 'index.html'. That is the actual file that you have called in when you first typed in 'localhost' after installing Apache2. After juno.php has been deposited into the var folder in www, go back to the web browser and type localhost/juno.php into the address bar and you should see the last name, a comma and the first name of the person that you entered. At this point the large prerequisite packages have been installed and it is time to move onto configuring Apache2 and MySQL for the Koha installation.

Appendix C: Scholarly Koha Resources

Bissels, G., & Chandler, A. (2010). Two years on: Koha 3.0 in use at the CAMLIS library, Royal London Homoeopathic Hospital. *Electronic Library and Information Services* , 44 (3), 283-290.

Canonical Ltd. (2011, 10 13). *Installation/SystemRequirements - Community Ubuntu Documentation*. Retrieved 11 30, 2011, from Ubuntu Documentation:

<https://help.ubuntu.com/community/Installation/SystemRequirements>

Canonical Ltd. (2011, 02 10). *Taskset - Community Ubuntu Documentation*. Retrieved 11 30, 2011, from Community Ubuntu Documentation: <https://help.ubuntu.com/community/Taskset>

Coar, K., & Bowen, R. (2008). *Apache Cookbook*. Sebastopol, CA: O'Reilly.

Dimant, N. (2010). Breaking the barriers: the role of support companies in making open source a reality. *Library Review* , 59 (9), 662-666.

FERRARO, J. (2009). LIBLIME LAUNCHES LIBLIME ENTERPRISE KOHA. *Advanced Technology Libraries*; , 2-2.

Helling, J. (2010). Cutting the proprietary cord: A case study fo one library's decision to migrate to an open source ILS. *Library Review* , 59 (9), 702-707.

Helling, J. (2010). Cutting the Proprietary Cord: A Case Study of One Library's Decision to Migrate to an Open Source ILS. *Library Review* , 59 (9), 702-707.

Helmke, M., Hudson, A., & Hudson, P. (2011). *Ubuntu: Unleashed*. Indianapolis: Sams.

Jones, P. (2011, 11 30). Personal Correspondence (email).

Koha Library Software Community. (2011). *Horowhenua Library Trust Koha Committee Rules | Koha Library Software Community*. Retrieved 11 30, 2011, from Koha-Community.org: <http://koha-community.org/about/koha-project-organization/horowhenua-library-trust-koha-committee-rules/>

Kohn, K., & McCloy, E. (2010). Phased Migration to Koha: Our Library's Experience. *Journal of Web Librarianship*, 427-434.

LibLime is the global leader in Koha support. (2011). Retrieved 11 30, 2011, from LibLime: Premier Open Source Library Support: <http://www.liblime.com/welcome>

Liblime. (2011). *Koha - Open Source ILS - Integrated Library System*. Retrieved 12 30, 2011, from Liblime: <http://www.koha.org/liblime-and-waldo-announce-that-liblime-academic-koha-48-is-now-in-production-release-and-is-already-in-use-by-dozens-of-liblime-academic-library-customers-?a=1&c=1254>

Liblime. (2011). *LibLime is the global leader in Koha support*. Retrieved 11 30, 2011, from Liblime: view-source:<http://www.liblime.com/contact>

Molyneux, R. E. (2011). An Open Source ILS Glossary: Version 2. *Public Library Quarterly*, 30 (2), 165-176.

Morton-Owens, E. G., Hanson, K. L., & Walls, I. (2011). Implementing Open-Source Software for Three Core Library Functions: A Stage-by-Stage Comparison. *Journal of Electronic Resources in Medical Libraries*, 8 (1), 1-14.

Muller, T. (2011). How to choose a free and open source integrated library system. *OCLC Systems & Services: International digital library perspectives*, 27 (1), 57-78.

phpMyAdmin devel team. (2011). *phpMyAdmin*. Retrieved 11 30, 2011, from phpMyAdmin: http://www.phpmyadmin.net/home_page/license.php

Sirohi, S., & Gupta, A. (2010). *Koha 3 Library Management System*. Birmingham, UK: PACKT.

Software in the Public Interest, Inc. (2011, 11 23). *dpkg - Debian Wiki*. Retrieved 11 30, 2011, from Debian: <http://wiki.debian.org/dpkg>

The Apache Software Foundation. (2011). *mod_rewrite - Apache HTTP Server*. Retrieved 11 30, 2011, from The Apache Software Foundation: http://httpd.apache.org/docs/current/mod/mod_rewrite.html

Walls, I. (2011). Migrating from Innovative Interfaces' Millennium to Koha. *OCLC Systems & Services: International digital library perspectives*, 27 (1), 51-56.

Yang, S., & Hofmann, M. (2010). The Next Generation Library Catalog: A Comparative Study of the OPACs of Koha, Evergreen, and Voyager. *Information Technology and Libraries*, 141-150.

Ziff Davis, Inc. (2011). *The Massive Windows 8 Tablet Wave Is Coming | News & Opinion | PCMag.com*. Retrieved 11 30, 2011, from PCMag.com: <http://www.pcmag.com/article2/0,2817,2393229,00.asp>

Appendix D: Quick Guide to Installing Koha

Quick Installation Sheet

Install Ubuntu Linux

Install Updates

Access root user

```
sudo passwd root
```

Install Lamp

Install tasksel

```
apt-get install tasksel
```

```
tasksel install lamp-server
```

here you create MySQL root password

install phpMyAdmin and MySQL Admin with package manager

Download Koha

```
apt-get install git
```

change directory to koha-user's home directory

```
cd /
```

```
cd home/koha
```

```
git clone git://git.koha-community.org/koha.git
```

rename new folder 'koha' to 'koha-clone'

Perl Modules

Automatic

```
apt-get install dselect
```

enter new Koha folder

```
cd /
```

```
cd home/koha/koha-clone
```

```
dpkg --get-selections < install_misc/Ubuntu.packages
```

```
sudo dselect
```

```
Install
```

```
Config
```

```
Remove
```

```
Quit
```

Manual

```
cpan
```

```
install Gravatar::URL
```

and the rest of these modules

```
Locale::Currency::Format
```

```
Memoize::Memcached
```

PDF::API2::Simple
Text::CSV::Encoded
Business::ISBN
CGI::Session::Driver::memcached
HTTP::OAI

MySQL setup

create user 'kohaadmin' and 'kohauser' in MySQL Admin

Username: kohaadmin

Password: katikoan

Username: kohauser

Password: zebrastripes

login to mysql

```
mysql -u root -p
```

```
create database koha;
```

```
grant all on koha.* to 'kohaadmin'@'localhost' identified by 'katikoan';
```

```
grant all on koha.* to 'kohauser'@'localhost' identified by 'katikoan';
```

```
flush privileges;
```

```
quit;
```

Real Installation

use terminal in koha-clone directory

```
perl Makefile.PL
```

answer all questions and check that all packages are installed

```
make
```

```
make test
```

```
sudo make install
```

Post Installation

configure apache2's ports.conf and http.conf changing each entry to 'localhost'

```
a2enmod rewrite
```

```
service apach2 restart
```

Appendix E: Password Cheat Sheet

Password Cheat Sheet

Administrative Password _____

Root Password _____

MySQL root password _____

phpMyAdmin password _____

Koha Admin

-username _____

-password _____

Koha User

-username _____

-password _____

Works Cited

- Bissels, G., & Chandler, A. (2010). Two years on: Koha 3.0 in use at the CAMLIS library, Royal London Homoeopathic Hospital. *Electronic Library and Information Services* , 44 (3), 283-290.
- Canonical Ltd. (2011, 10 13). *Installation/SystemRequirements - Community Ubuntu Documentation*. Retrieved 11 30, 2011, from Ubuntu Documentation: <https://help.ubuntu.com/community/Installation/SystemRequirements>
- Canonical Ltd. (2011, 02 10). *Taskel - Community Ubuntu Documentation*. Retrieved 11 30, 2011, from Community Ubuntu Documentation: <https://help.ubuntu.com/community/Taskel>
- Coar, K., & Bowen, R. (2008). *Apache Cookbook*. Sebastopol, CA: O'Reilly.
- Dimant, N. (2010). Breaking the barriers: the role of support companies in making open source a reality. *Library Review* , 59 (9), 662-666.
- FERRARO, J. (2009). LIBLIME LAUNCHES LIBLIME ENTERPRISE KOHA. *Advanced Technology Libraries*; , 2-2.
- Helling, J. (2010). Cutting the proprietary cord: A case study fo one library's decision to migrate to an open source ILS. *Library Review* , 59 (9), 702-707.
- Helling, J. (2010). Cutting the Proprietary Cord: A Case Study of One Library's Decision to Migrate to an Open Source ILS. *Library Review* , 59 (9), 702-707.
- Helmke, M., Hudson, A., & Hudson, P. (2011). *Ubuntu: Unleashed*. Indianapolis: Sams.
- Jones, P. (2011, 11 30). Personal Correspondence (email).
- Koha Library Software Community. (2011). *Horowhenua Library Trust Koha Committee Rules | Koha Library Software Community*. Retrieved 11 30, 2011, from Koha-Community.org: <http://koha-community.org/about/koha-project-organization/horowhenua-library-trust-koha-committee-rules/>
- Kohn, K., & McCloy, E. (2010). Phased Migration to Koha: Our Library's Experience. *Journal of Web Librarianship* , 427-434.
- LibLime is the global leader in Koha support*. (2011). Retrieved 11 30, 2011, from LibLime: Premier Open Source Library Support: <http://www.liblime.com/welcome>
- Liblime. (2011). *Koha - Open Source ILS - Integrated Library System*. Retrieved 12 30, 2011, from Liblime: <http://www.koha.org/liblime-and-waldo-announce-that-liblime-academic-koha-48-is-now-in-production-release-and-is-already-in-use-by-dozens-of-liblime-academic-library-customers-?a=1&c=1254>
- Liblime. (2011). *LibLime is the global leader in Koha support*. Retrieved 11 30, 2011, from Liblime: view-source:<http://www.liblime.com/contact>

Molyneux, R. E. (2011). An Open Source ILS Glossary: Version 2. *Public Library Quarterly* , 30 (2), 165-176.

Morton-Owens, E. G., Hanson, K. L., & Walls, I. (2011). Implementing Open-Source Software for Three Core Library Functions: A Stage-by-Stage Comparison. *Journal of Electronic Resources in Medical Libraries* , 8 (1), 1-14.

Muller, T. (2011). How to choose a free and open source integrated library system. *OCLC Systems & Services: International digital library perspectives* , 27 (1), 57-78.

phpMyAdmin devel team. (2011). *phpMyAdmin*. Retrieved 11 30, 2011, from phpMyAdmin: http://www.phpmyadmin.net/home_page/license.php

Sirohi, S., & Gupta, A. (2010). *Koha 3 Library Management System*. Birmingham, UK: PACKT.

Software in the Public Interest, Inc. (2011, 11 23). *dpkg - Debian Wiki*. Retrieved 11 30, 2011, from Debian: <http://wiki.debian.org/dpkg>

The Apache Software Foundation. (2011). *mod_rewrite - Apache HTTP Server*. Retrieved 11 30, 2011, from The Apache Software Foundation: http://httpd.apache.org/docs/current/mod/mod_rewrite.html

Walls, I. (2011). Migrating from Innovative Interfaces' Millennium to Koha. *OCLC Systems & Services: International digital library perspectives* , 27 (1), 51-56.

Yang, S., & Hofmann, M. (2010). The Next Generation Library Catalog: A Comparative Study of the OPACs of Koha, Evergreen, and Voyager. *Information Technology and Libraries* , 141-150.

Ziff Davis, Inc. (2011). *The Massive Windows 8 Tablet Wave Is Coming | News & Opinion | PCMag.com*. Retrieved 11 30, 2011, from PCMag.com: <http://www.pcmag.com/article2/0,2817,2393229,00.asp>